

Міністерство освіти і науки України
Одеський національний політехнічний університет
Інститут комп'ютерних систем
Кафедра інформаційних систем

Лабораторна робота №10
З дисципліни «Операційні системи»

Тема: «Керування процесами-транзакціями в базах даних. Частина 2»

Виконав:
Ст. гр. АІ-204
Дорожкін Михайло

Перевірив(-ла):
Блажко О. А.
Дрозд М.О.

Одеса 2021

Мета роботи: дослідити поведінку процесів-транзакцій в базах даних та засоби керування ними через механізм блокування з використанням сучасних систем керування базами даних.

2 Завдання

Для кожної транзакції підготуйте окремий термінал, в якому виконайте команду доступу до вашої БД з використанням утиліти `psql`.

Завдання 1. Аналіз роботи багато версійного протоколу

В завданні 1 рішення попередньої лабораторної роботи було створено таблицю з декількома рядками.

Підготуйте чотири транзакції за прикладом з рисунку 2:

- T1 – отримання номеру транзакції, внесення нового рядка в таблицю та перегляд вмісту таблиці;
- T2 – постійний перегляд вмісту таблиці
- T3 – видалення рядку з наступною відміною цієї операції;
- T4 – зміна значення однієї з колонок рядка.

В операцію читання рядка таблиці додайте системні колонки `xmin`, `xmax`.

На кожному кроці виконання транзакції переглядайте значення колонок `xmin`, `xmax` та зробіть відповідні висновки.

Завдання 2. Аналіз стану транзакцій на різних рівнях багаторівневого блокування

Виконайте послідовно в двох терміналах наступні комбінації блокувань таблиці: IX-IS, SIX-IX, SIX-IS. Надайте висновки про сумісність блокувань.

Для кожної комбінації блокувань перед завершенням 1-ї транзакції (яка розпочалася раніше) в додатковому терміналі через команду `psql` отримайте данні про стан транзакцій (таблиця `pg_locks`).

Завдання 3. Керування квазіпаралельним виконанням транзакцій на різних рівнях ізоляції транзакцій

Підготуйте транзакції, які було створено у завданні 3.1 рішення попередньої лабораторної роботи, а саме, створіть дві транзакції, кожна з яких повинна включати такі операції:

- операція читання першого рядку таблиці;

- операція редагування однієї із змінних таблиці в першому рядку;
- повторна операція читання першого рядку таблиці;
- операція фіксації всіх змін.

1.1 Виконайте роботу транзакцій при умові їх роботи на рівні ізоляції READ COMMITTED. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

1.2 Повторіть роботу транзакцій при умові їх роботи на рівні ізоляції REPEATABLE READ. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

1.3 Повторіть роботу транзакцій при умові їх роботи на рівні ізоляції SERIALIZABLE. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

Завдання 4. Керування квазіпаралельним виконанням транзакцій при наявності тупикових ситуацій.

3.1 Виконайте модифікацію транзакцій так, щоб вони призводили до тупикової ситуації.

3.2 Виконайте дві модифіковані транзакції.

Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та яка призвела до тупику. Дайте свої висновки з урахуванням:

- ідентифікаторів процесів
- номерів транзакцій.

Виконання завдань:

Завдання 1. Аналіз роботи багатоверсійного протоколу

В завданні 1 рішення попередньої лабораторної роботи було створено таблицю з декількома рядками. В операцію читання рядка таблиці додайте системні колонки xmin, xmax.

На кожному кроці виконання транзакції переглядайте значення колонок xmin, xmax та зробіть відповідні висновки.

Підготуйте чотири транзакції за прикладом з рисунку 2:

- T1 – отримання номеру транзакції, внесення нового рядка в таблицю та перегляд вмісту таблиці;
- T2 – постійний перегляд вмісту таблиці

```
dorozhkin_mihajlo@vpsj3IeQ:~  
login as: dorozhkin_mihajlo  
dorozhkin_mihajlo@91.219.60.189's password:  
Last login: Tue May 4 20:37:42 2021 from 83.142.111.242  
[dorozhkin_mihajlo@vpsj3IeQ ~]$ start transaction;  
-bash: start: command not found  
[dorozhkin_mihajlo@vpsj3IeQ ~]$ psql  
psql (9.5.25)  
Type "help" for help.  
  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> INSERT INTO worker VALUES (3, 'Alekseev', '01/12/1994');  
INSERT 0 1  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
 xmin | xmax | name  
-----  
 2140 |    0 | Petrov  
 2179 |    0 | Ivanov  
 3019 |    0 | Alekseev  
(3 rows)  
  
dorozhkin_mihajlo=> COMMIT;  
COMMIT  
dorozhkin_mihajlo=>   
  
dorozhkin_mihajlo@vpsj3IeQ:~  
login as: dorozhkin_mihajlo  
dorozhkin_mihajlo@91.219.60.189's password:  
Last login: Tue May 4 20:42:48 2021 from 83.142.111.242  
[dorozhkin_mihajlo@vpsj3IeQ ~]$ psql  
psql (9.5.25)  
Type "help" for help.  
  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
 xmin | xmax | name  
-----  
 2140 |    0 | Petrov  
 2179 |    0 | Ivanov  
(2 rows)  
  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
 xmin | xmax | name  
-----  
 2140 |    0 | Petrov  
 2179 |    0 | Ivanov  
(2 rows)  
  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
 xmin | xmax | name  
-----  
 2140 |    0 | Petrov  
 2179 |    0 | Ivanov  
 3019 |    0 | Alekseev  
(3 rows)  
  
dorozhkin_mihajlo=>   

```

– T3 – видалення рядку з наступною відміною цієї операції;

```
dorozhkin_mihajlo@vpsj3leQ:~  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> DELETE FROM worker WHERE p_id = 3;  
DELETE 1  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
(2 rows)  
  
dorozhkin_mihajlo=> ABORT;  
ROLLBACK  
dorozhkin_mihajlo=>   
  
dorozhkin_mihajlo@vpsj3leQ:~  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
(2 rows)  
  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
3051 | 0 | Alekseev  
(3 rows)  
  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
3051 | 3052 | Alekseev  
(3 rows)
```

– T4 – зміна значення однієї з колонок рядка.

```
dorozhkin_mihajlo@vpsj3leQ:~  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> UPDATE worker SET name = 'Aleksov' WHERE p_id = 3;  
UPDATE 1  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
3054 | 0 | Aleksov  
(3 rows)  
  
dorozhkin_mihajlo=> COMMIT;  
COMMIT  
dorozhkin_mihajlo=>   
  
dorozhkin_mihajlo@vpsj3leQ:~  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
3051 | 3054 | Alekseev  
(3 rows)  
  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
3051 | 3054 | Alekseev  
(3 rows)  
  
dorozhkin_mihajlo=> SELECT xmin,xmax, name FROM worker;  
xmin | xmax | name  
-----  
2140 | 0 | Petrov  
2179 | 0 | Ivanov  
3054 | 0 | Aleksov  
(3 rows)
```

Завдання 2. Аналіз стану транзакцій на різних рівнях багаторівневого блокування

Виконайте послідовно в двох терміналах наступні комбінації блокувань таблиці:

IX-IS:

```
dorozhkin_mihajlo@vpsj3leQ:~  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> LOCK TABLE worker IN ROW EXCLUSIVE MODE;  
LOCK TABLE  
dorozhkin_mihajlo=>   
  
dorozhkin_mihajlo@vpsj3leQ:~  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> LOCK TABLE worker IN ROW SHARE MODE;  
LOCK TABLE  
dorozhkin_mihajlo=>   
  
dorozhkin_mihajlo@vpsj3leQ:~  
relation | locktype | virtualtransaction | pid | mode | granted  
-----  
11673 | relation | 23/1393 | 30607 | AccessShareLock | t  
16759 | relation | 12/15833 | 21626 | AccessShareLock | t  
11673 | relation | 38/160 | 3342 | AccessShareLock | t  
--More--
```

SIX-IX:

```
dorozhkin_mihajlo@vpsj3leQ:~  
login as: dorozhkin_mihajlo  
dorozhkin_mihajlo@91.219.60.189's password:  
Last login: Wed May 5 09:43:22 2021 from 83.142.111.242  
[dorozhkin_mihajlo@vpsj3leQ ~]$ psql  
psql (9.5.25)  
Type "help" for help.  
  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> lock table worker in share row exclusive mode;  
LOCK TABLE  
dorozhkin_mihajlo=>   
  
dorozhkin_mihajlo@vpsj3leQ:~  
login as: dorozhkin_mihajlo  
dorozhkin_mihajlo@91.219.60.189's password:  
Last login: Wed May 5 12:25:08 2021 from 83.142.111.242  
[dorozhkin_mihajlo@vpsj3leQ ~]$ psql  
psql (9.5.25)  
Type "help" for help.  
  
dorozhkin_mihajlo=> start transaction;  
START TRANSACTION  
dorozhkin_mihajlo=> lock table worker in row exclusive mode;  
LOCK TABLE  
dorozhkin_mihajlo=>   
  
5  
акц  
исут  
табл  
- в  
блок  
заб.  
  
dorozhkin_mihajlo@vpsj3leQ:~  
dorozhkin_mihajlo=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_locks where locktype = 'relation';  
relation | locktype | virtualtransaction | pid | mode | granted  
-----  
16837 | relation | 11/10860 | 19478 | RowExclusiveLock | t  
11673 | relation | 9/22717 | 19228 | AccessShareLock | t  
11673 | relation | 28/1781 | 11516 | AccessShareLock | t  
11673 | relation | 21/5121 | 29192 | AccessShareLock | t  
11673 | relation | 4/109855 | 8604 | AccessShareLock | t  
16636 | relation | 8/29643 | 19221 | RowExclusiveLock | f  
16636 | relation | 6/74050 | 19213 | ShareRowExclusiveLock | t  
16663 | relation | 21/5121 | 29192 | ShareRowExclusiveLock | t  
(8 rows)
```

SIX-IS:

The screenshot shows two terminal windows. The left window shows a sequence of commands: `start transaction;`, `lock table worker in share row exclusive mode;`, and `LOCK TABLE`. The right window shows the same sequence but with `lock table worker in row share mode;`. Below these, a third window displays the output of `SELECT * FROM pg_locks;` as a table.

relation	locktype	virtualtransaction	pid	mode	granted
16837	relation	11/10860	19478	RowExclusiveLock	t
16705	relation	12/19084	19822	AccessShareLock	t
16705	relation	12/19084	19822	RowExclusiveLock	t
11673	relation	9/22718	19228	AccessShareLock	t
11673	relation	28/1781	11516	AccessShareLock	t
11673	relation	21/5121	29192	AccessShareLock	t
16837	relation	10/12628	19461	RowShareLock	t
16705	relation	13/8335	19876	AccessShareLock	t
11673	relation	4/109855	8604	AccessShareLock	t
16636	relation	8/29644	19221	RowShareLock	t
16636	relation	6/74052	19213	ShareRowExclusiveLock	t

Надайте висновки про сумісність блокувань.

Можна зробити висновок, що тільки блоки SIX-IX не є сумісними з наведених трьох прикладів.

Для кожної комбінації блокувань перед завершенням 1-ї транзакції (яка розпочалася раніше) в додатковому терміналі через команду `psql` отримайте данні про стан транзакцій (таблиця `pg_locks`).

Завдання 3. Керування квазіпаралельним виконанням транзакцій на різних рівнях ізоляції транзакцій

Підготуйте транзакції, які було створено у завданні 3.1 рішення попередньої лабораторної роботи, а саме, створіть дві транзакції, кожна з яких повинна включати такі операції:

- операція читання першого рядку таблиці;
- операція редагування однієї із змінних таблиці в першому рядку;
- повторна операція читання першого рядку таблиці;
- операція фіксації всіх змін.

Транзакція 1	Транзакція 2
START TRANSACTION	START TRANSACTION
SELECT * FROM worker WHERE p_id = 1;	SELECT * FROM worker WHERE p_id = 1;
UPDATE worker SET name = 'Puzanov' WHERE p_id = 1;	UPDATE worker SET bd = '01/01/2005' WHERE p_id = 1;
SELECT * FROM worker WHERE p_id = 1;	SELECT * FROM worker WHERE p_id = 1;
COMMIT;	COMMIT;

1.1 Виконайте роботу транзакцій при умові їх роботи на рівні ізоляції READ COMMITTED. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

```

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> start transaction;
START TRANSACTION
dorozhkin_mihajlo=> SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
SET
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Aleksov | 1994-01-12
(3 rows)

dorozhkin_mihajlo=> UPDATE worker SET name = 'Puzanov' WHERE p_id = 3;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanov | 1994-01-12
(3 rows)

dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=>

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> start transaction;
START TRANSACTION
dorozhkin_mihajlo=> SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
SET
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Aleksov | 1994-01-12
(3 rows)

dorozhkin_mihajlo=> UPDATE worker SET bd = '01/01/2005' WHERE p_id = 3;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanov | 2005-01-01
(3 rows)

dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=>

```

1.2 Повторіть роботу транзакцій при умові їх роботи на рівні ізоляції REPEATABLE READ. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

```

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> start transaction;
START TRANSACTION
dorozhkin_mihajlo=> SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;
SET
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanov | 2005-01-01
(3 rows)

dorozhkin_mihajlo=> UPDATE worker SET name = 'Puzanich' WHERE p_id = 3;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanich | 2005-01-01
(3 rows)

dorozhkin_mihajlo=> COMMIT;
COMMIT

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> start transaction;
START TRANSACTION
dorozhkin_mihajlo=> SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;
SET
dorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanov | 2005-01-01
(3 rows)

dorozhkin_mihajlo=> UPDATE worker SET bd = '03/03/2003' WHERE p_id = 3;
ERROR: could not serialize access due to concurrent update
dorozhkin_mihajlo=>
dorozhkin_mihajlo=> UPDATE worker SET bd = '03/03/2003' WHERE p_id = 3;
ERROR: current transaction is aborted, commands ignored until end of transaction block
dorozhkin_mihajlo=>

```


1.3 Повторіть роботу транзакцій при умові їх роботи на рівні ізоляції SERIALIZABLE. Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та дайте свої висновки.

```
dorozhkin_mihajlo@vpsj3leQ:~
lorozhkin_mihajlo=> start transaction;
START TRANSACTION
lorozhkin_mihajlo=> SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
SET
lorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanich | 2005-01-01
(3 rows)

lorozhkin_mihajlo=> UPDATE worker SET name = 'Puzanov' WHERE p_id = 3;
UPDATE 1
lorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanov | 2005-01-01
(3 rows)

lorozhkin_mihajlo=> COMMIT;
COMMIT
lorozhkin_mihajlo=> 
```

```
dorozhkin_mihajlo@vpsj3leQ:~
lorozhkin_mihajlo=> start transaction;
START TRANSACTION
lorozhkin_mihajlo=> SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
SET
lorozhkin_mihajlo=> SELECT * FROM worker;
p_id | name | bd
-----+-----+-----
2 | Petrov | 2001-04-01
1 | Ivanov | 2001-05-04
3 | Puzanich | 2005-01-01
(3 rows)

lorozhkin_mihajlo=> UPDATE worker SET bd = '03/03/2003' WHERE p_id = 3;
ERROR: could not serialize access due to concurrent update
lorozhkin_mihajlo=> SELECT * FROM worker;
ERROR: current transaction is aborted, commands ignored until end of transaction block
lorozhkin_mihajlo=> 
```

Завдання 4. Керування квазіпаралельним виконанням транзакцій при наявності тупикових ситуацій.

3.1 Виконайте модифікацію транзакцій так, щоб вони призводили до тупикової ситуації.

Транзакція 1	Транзакція 2
START TRANSACTION	START TRANSACTION
UPDATE worker SET name = 'Puzanov' WHERE p_id = 3;	UPDATE worker SET bd = '03/03/2003' WHERE p_id = 2;
UPDATE worker SET bd = '01/01/2005' WHERE p_id = 2;	UPDATE worker SET name = 'Puzanich' WHERE p_id = 3;

3.2 Виконайте дві модифіковані транзакції.

Проаналізуйте реакцію СКБД на операцію UPDATE 2-ї транзакції (яка виконується пізніше) та яка призвела до тупику. Дайте свої висновки з урахуванням:

- ідентифікаторів процесів
- номерів транзакцій.

```
dorozhkin_mihajlo@vpsj3IeQ:~
dorozhkin_mihajlo=> start transaction;
START TRANSACTION
dorozhkin_mihajlo=> UPDATE worker SET name = 'Puzanov' WHERE p_id = 3;
UPDATE 1
dorozhkin_mihajlo=> UPDATE worker SET name = '03/01/1987' WHERE p_id = 2;
UPDATE 1
dorozhkin_mihajlo=>

dorozhkin_mihajlo@vpsj3IeQ:~
dorozhkin_mihajlo=> start transaction;
START TRANSACTION
dorozhkin_mihajlo=> UPDATE worker SET bd = '12/12/2005' WHERE p_id = 2;
UPDATE 1
dorozhkin_mihajlo=> UPDATE worker SET name = 'Aleksovich' WHERE p_id = 3;
ERROR: deadlock detected
DETAIL: Process 25731 waits for ShareLock on transaction 3420; blocked by process 19213.
Process 19213 waits for ShareLock on transaction 3421; blocked by process 25731.
HINT: See server log for query details.
CONTEXT: while updating tuple (0,18) in relation "worker"
dorozhkin_mihajlo=>

dorozhkin_mihajlo@vpsj3IeQ:~
For more details see ps(1).
[dorozhkin_mihajlo@vpsj3IeQ ~]$ ps -u postgres -o pid,ppid,stat,cmd | egrep "dorozhkin_mihajlo"
error: user name does not exist

Usage:
ps [options]

Try 'ps --help <simple|list|output|threads|misc|all>'
or 'ps --help <s|l|o|t|m|a>'
for additional help text.

For more details see ps(1).
[dorozhkin_mihajlo@vpsj3IeQ ~]$ ps -u postgres -o pid,ppid,stat,cmd | egrep "dorozhkin_mihajlo"
25731  8763 Ss   postgres: dorozhkin_mihajlo dorozhkin_mihajlo [local] idle
[dorozhkin_mihajlo@vpsj3IeQ ~]$ ps -u postgres -o pid,ppid,stat,cmd | egrep "dorozhkin_mihajlo"
25731  8763 Ss   postgres: dorozhkin_mihajlo dorozhkin_mihajlo [local] idle
26934  8763 Ss   postgres: dorozhkin_mihajlo dorozhkin_mihajlo [local] idle in transaction
26993  8763 Ss   postgres: dorozhkin_mihajlo dorozhkin_mihajlo [local] idle in transaction (aborted)
[dorozhkin_mihajlo@vpsj3IeQ ~]$
```

Висновок: Під час виконання лабораторної роботи ми дослідили поведінку процесів-транзакцій в базах даних та засоби керування ними через механізм блокування з використанням сучасних систем керування базами даних.